

## REMARKS

### **Overview of the Office Action**

Claims 1, 2, 8, 9, 15-20, and 22-27 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Coad et al. (U.S. Patent No. 6,851,107) in view of Dye et al. in further view of Banning et al. (U.S. Patent No. 5,485,567) and in further view of Rivlin (U.S. Patent 6,032,159).

Claims 3-7, 21, and 28 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Coad et al. in view of Banning, and further in view of Rivlin, and further in view of Washburn et al. (U.S. Patent No. 5,157,779).

Claim 12 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Coad et al. in view of Dye, in further view of Banning, in further view of Rivlin, and in further view of Peddada et al. (U.S. Patent No. 6,031,533).

Claim 13 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Coad et al. in view of Dye, in further view of Banning, in further view of Rivlin, and in further view of Gupta et al. (U.S. Patent No. 6,484,156).

Claim 14 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Coad et al. in view of Dye, in further view of Banning, in further view of Rivlin, in further view of Gupta, and in further view of O'Donnell et al. (U.S. Patent No. 6,223,203).

### **Status of the Claims/Amendments**

Claims 1-9 and 12-28 are pending.

### **Claims Rejected Under 35 U.S.C. § 103(a)**

Claims 1, 15, and 22 include features that are neither disclosed nor suggested by the art of record, namely, as represented by claim 1:

A computerized system for software development comprising:

a source code editor operable to edit a source code module;

a graphical design surface operable to display a graphical object representing actual code of the source code module **and provide integrated testing of the source code module**; and

wherein upon a change in the source code module, the change in the source code is immediately communicated to the graphical design surface and the graphical design surface is updated to reflect the change in the source code module, wherein the design surface displays the graphical object, the graphical object represents a database object, the design surface is operative to bind a particular database system to the database object, the database object further includes a database column, the source code module includes a variable, and the design surface is operative to bind the database column to the variable.

Claim 1 is directed to a system that comprises a source code editor that can edit a source code module, and a graphical design surface that can display a graphical object representing code of the source code module. The graphical design surface displays a database object, including a database field. The graphical design surface can bind the database field to a variable of the source code module. In addition, the graphical design surface includes the ability to test the source code module through the graphical design surface.

Claim 1 includes the feature of the graphical design surface including the ability to test the source code module through the graphical design surface. Support for this feature can be found on page 21 of the specification as originally filed.

Coad et al. teaches a software development tool that provides the simultaneous view of a graphical and text version of source code. The views are synchronized so that modification of one view is reflected in the other. However, there is no teaching of integrated testing of the source code through the design surface, as required by the claim.

Dye teaches a system for enabling the distributed display of a user interface of a graphical program executing on a server computer. As the program on the server executes, client computers display one or more representations of graphical user interfaces of the graphical program. For example, the graphical user interfaces may be virtual instruments corresponding to control panels of a factory process. When the associated program executes on the server, the client computers may each display one of the virtual instrument panels corresponding to each of the control panels that control the process. However, there is no mention of integrated testing of source code through a design surface anywhere in Dye.

Banning et al. teaches a system for displaying database information in a table using icons to convey properties of the database to a user. There is no mention of integrated testing of source code through a design surface anywhere in Banning et al.

Rivlin teaches a method for automatically copying files to a database. There is no mention of a integrated testing of the source code through the design surface anywhere in Rivlin.

Washburn et al. teaches a system for comparing and identifying differences between a first piece of data and a second piece of data. The identified differences are then displayed to the user. There is no mention of a integrated testing of the source code through the design surface anywhere in Washburn et al.

Gupta et al. teaches the use of a hierarchical annotation storage structure to maintain a correspondence between a plurality of multimedia stream annotations and a hierarchical higher group identifier. There is no mention of a integrated testing of the source code through the design surface anywhere in Gupta et al.

Peddada et al. teaches a method for providing a graphical user interface to a client device. There is no mention of a integrated testing of the source code through the design surface anywhere in Peddada et al.

O'Donnell et al. teaches a system for executing a process in parallel across a plurality of networked computers. There is no mention of a integrated testing of the source code through the design surface anywhere in O'Donnell et al.

The Examiner states that Dye teaches integrated testing of the source code module by the design surface at ¶ 12. Applicant respectfully disagrees. The cited portion of Dye merely describes how it would be desirable to separate the front panels, or graphical user interfaces, from the block diagram. The cited portion further states that it would be desirable to have these separated panels be displayed on separate workstations to simulate how these panels might look in the real world.

This is not the same as integrated testing of the source code module by the design surface. First, there is no mention of testing the source code module anywhere in Dye. While the cited portion describes a "test and measurement application" this is in the context of describing what the application does, and has nothing to do with the capabilities of the design surface.

Second, even if testing the source code module was disclosed, there is no integration of the testing into the design surface. As taught by Dye, the graphical user interfaces are separated from the design surface. Thus, there is no integrated testing, because the graphical user interfaces are separated, not integrated, into various client stations.

Further, there is no motivation to combine the teaching of Dye with the prior art of record, Dye is directed towards programming process control applications that involve multiple panel type input devices. In contrast, the systems of Coad et al., Banning et al, Rivlin, and Washburn et al. are directed to programming conventional software applications that typically only receive input from a single user through a mouse or other input device, and not the multiple panel type devices that are used in Dye. Accordingly, there would be no motivation to combine Dye with these systems because there would be no gain in displaying the graphical interfaces of the panel devices on separate clients. In fact, such an arrangement would complicate the testing because one would have to use a separate client computer to provide input to the design surface.

Because none of the cited art describes **integrated testing of the source code module by the design surface**, and there is no motivation to combine the teachings of Dye with the other cited prior art, Applicant respectfully requests that the Examiner withdraw the rejection and allow claim 1

Claims 15 and 22 include similar features as claim 1, and are therefore allowable for at least the reasons given above for claim 1. It is therefore respectfully requested that the Examiner withdraw the rejections and allow claims 15 and 22.

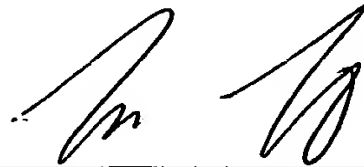
Claims 2-9, 12-14, 16-21, and 23-28 are all variously dependent on independent claims 1, 15, and 22, and are therefore allowable for at least the reasons given above for the independent claims. It is therefore respectfully requested that the Examiner withdraw the rejections and allow claims 2-9, 12-14, 16-21, and 23-28.

**DOCKET NO.:** MSFT-0556 /140707.01  
**Application No.:** 09/717,680  
**Office Action Dated:** April 10, 2006

**PATENT**  
**REPLY FILED UNDER EXPEDITED**  
**PROCEDURE PURSUANT TO**  
**37 CFR § 1.116**

In view of the foregoing remarks, Applicants submit that the above-identified application is in condition for allowance. Early notification to this effect is respectfully requested.

Date: June 7, 2006



---

Michael W. Tieff  
Registration No. 57,845

Woodcock Washburn LLP  
One Liberty Place - 46th Floor  
Philadelphia PA 19103  
Telephone: (215) 568-3100  
Facsimile: (215) 568-3439